Project 4 Task 1 – Tone Analyser App

Description:

My application takes an input text string from the user in the android app, and uses it to fetch and display the analysed tone scores for the text from IBM Watson API.

Here is how my application meets the task requirements
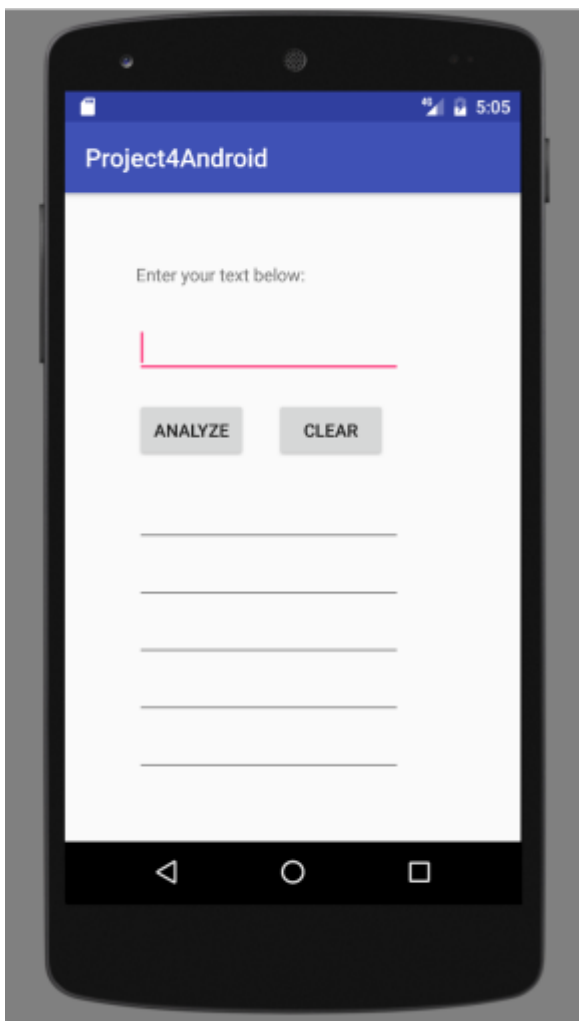
1. Implement a native Android application

The name of my native Android application project in Android Studio is: Project4Android

1.1. Has at least two different kinds of views in your Layout (TextView, EditText, ImageView, etc.)

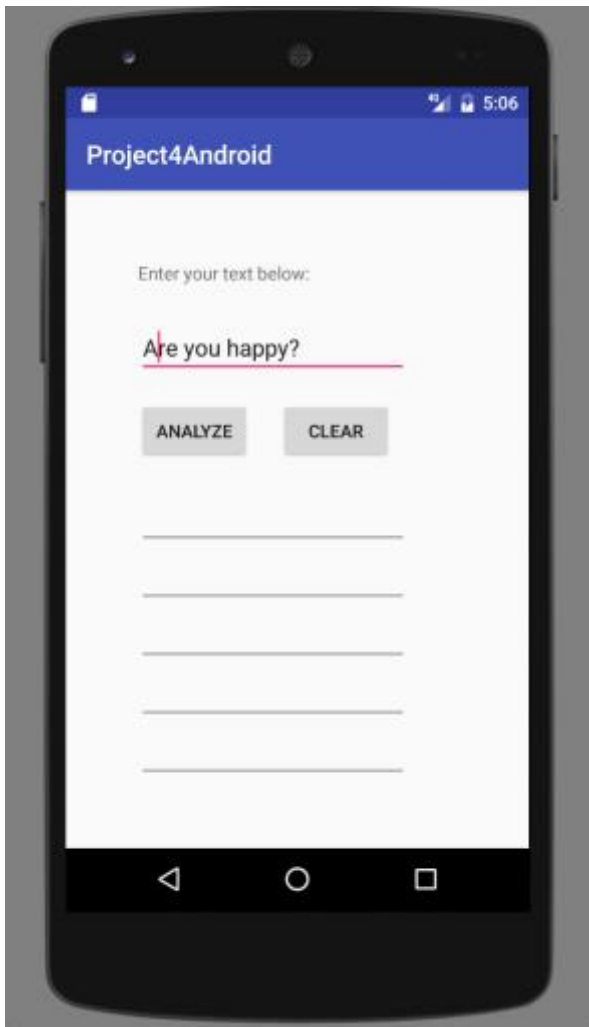My application uses TextView, EditText, and Button.
See activity_my_tone.xml for details of how they are incorporated into the LinearLayout.

Here is a screenshot of the layout before the picture has been fetched.

1.2. Requires input from the user

Here is a screenshot of the user fetching the tone scores for an input text string : "Are you happy?"

1.3. Makes an HTTP request (using an appropriate HTTP method) to your web service

My application does an HTTP GET request in GetEmotion.java. The HTTP request is:

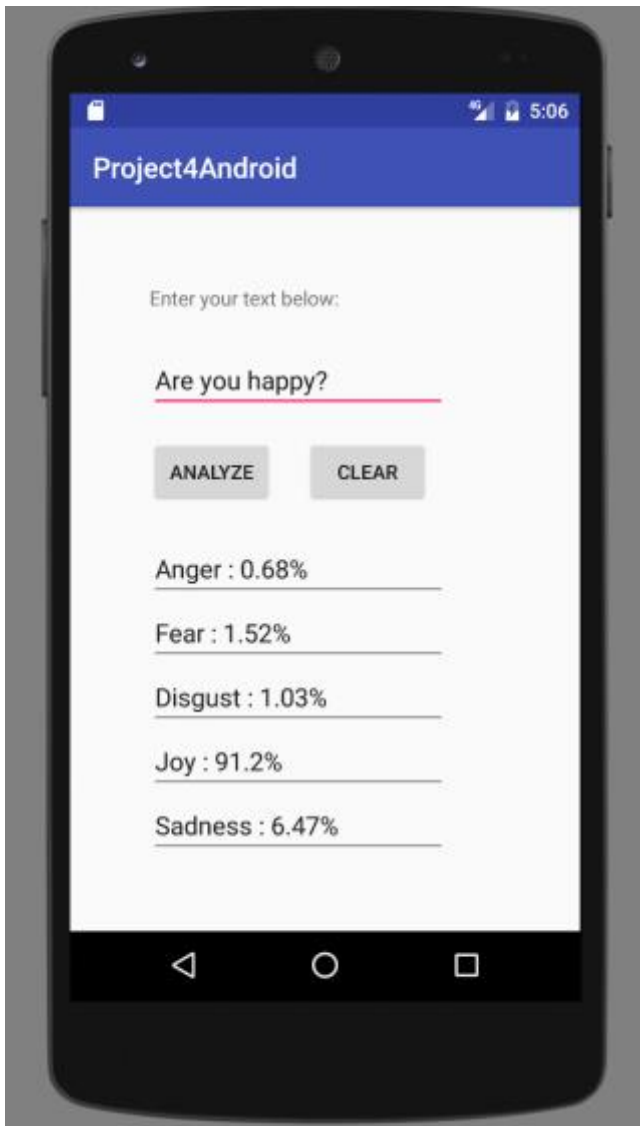**"https://agile-cove-45583.herokuapp.com//EmotionAnalyzer/" + analyzeText**

where analyzeText is the user's input text string. The analyse method makes this request of my web application, parses the returned JSON to fetch the tone scores, and returns the toneText of the input string.

1.4. Receives and parses JSON formatted reply from the web service

```
{"Emotion Tone": [
        {
        "Anger":"0.00672",
        "Fear":"0.015258",
        "Disgust":"0.010305",
        "Joy":"0.912158",
        "Sadness":"0.064564"
        }
]
}
```
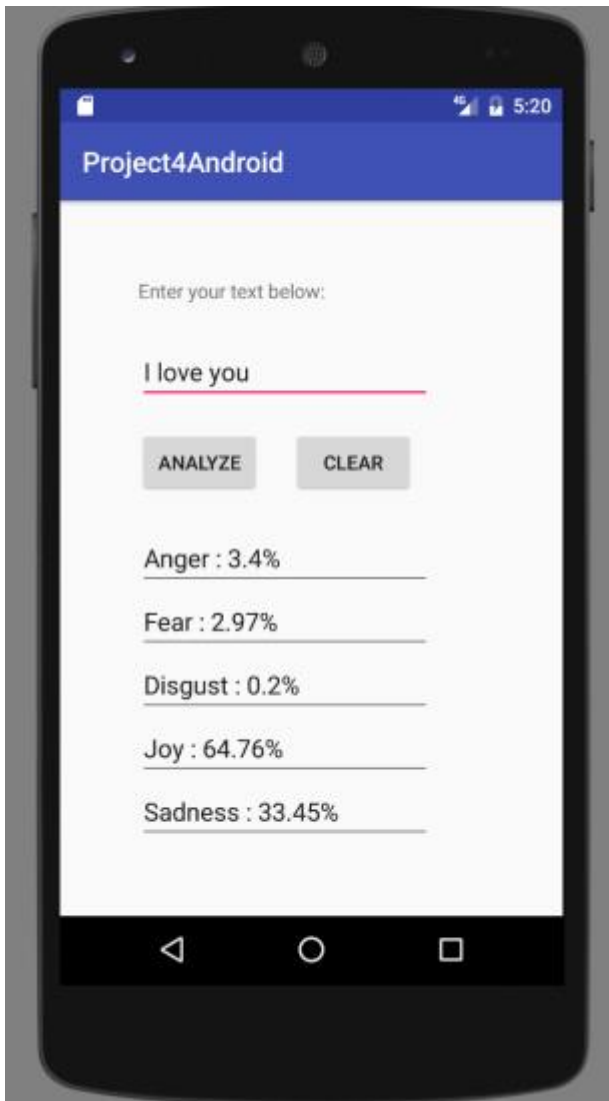
1.5. Displays new information to the user

Here is the screen shot after the picture has been returned.

1.6. Is repeatable (I.e. the user can repeatedly reuse the application without restarting it)

The user can type in another input text string and hit Analyze. Here is an example of having typed in "I love you". We can also clear the text for inputting the next text string by pressing the clear button.

2. Implement a web application, deployed to Heroku

The URL of my web service deployed to Heroku is:

agile-cove-45583.herokuapp.com

The project directory name is Project4Task1.

2.1. Using an HttpServlet to implement a simple API

In my web app project:
Model: ToneProcess.java
Controller: EmotionAnalyzer.java

2.2. Receives an HTTP request from the native Android application

EmotionAnalyzer.java receives the HTTP GET request with the argument "name". It passes this input string on to the model. Also, it makes an HTTP request to:

https://watson-api-explorer.mybluemix.net/tone-analyzer/api/v3/tone?version=2016-05-19&text="+name

2.3. Executes business logic appropriate to my application

ToneProcess.java parses the JSON response and extracts the parts it needs to respond to the Android application.

2.4. Replies to the Android application with a JSON formatted response.

The EmotionAnalyzer.java formats the extracted parts of the JSON response in JSON format to send it to the android application.

```
{"Emotion Tone": [
     {
     "Anger":"0.00672",
     "Fear":"0.015258",
     "Disgust":"0.010305",
     "Joy":"0.912158",
     "Sadness":"0.064564"
     }
]
}
```

Whereas response received from the IBM Watson API is:

```
{
```

```
"document_tone": {
  "tone_categories": [
    {
      "tones": [
        {
          "score": 0.553739,
          "tone_id": "anger",
          "tone_name": "Anger"
        },
        {
          "score": 0.057233,
          "tone_id": "disgust",
          "tone_name": "Disgust"
        },
        {
          "score": 0.121376,
          "tone_id": "fear",
          "tone_name": "Fear"
        },
        {
          "score": 0.025028,
          "tone_id": "joy",
          "tone_name": "Joy"
        },
        {
          "score": 0.306812,
          "tone_id": "sadness",
          "tone_name": "Sadness"
        }
      ],
      "category_id": "emotion_tone",
      "category_name": "Emotion Tone"
    }
  }
]}}
```